REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application. Claim 5 is canceled without prejudice. Claims 1-4 and 6-20 are pending in this application.

Specification

The specification stands objected to because the claims section does not start with "I (or we) claim," "The invention claimed is" (or the equivalent). As part of this response, this informality has been corrected. Accordingly, Applicant respectfully requests that the objection to the specification be withdrawn.

35 U.S.C. § 102

Claims 1, 3-5, 7, 15, 17, and 19 stand rejected under 35 U.S.C. §102(e) as being unpatentable over U.S. Patent Application Publication No. 20040003370A1 to Schenk et al. (hereinafter "Schenk"). Claim 5 has been canceled without prejudice, thereby rendering the rejection of claim 5 moot. Applicant respectfully submits that claims 1, 3, 4, 7, 15, 17, and 19 are not anticipated by Schenk.

Schenk is directed to systems and methods that optimize art asset rendering operations by using shader-driven compilation techniques (see, ¶ 19). As discussed in the Abstract of Schenk, pre-processing is performed in a compilation process. Geometric data are processed in the compilation process with knowledge of associated shading programs. The data are converted into data structures targeted directly to a target hardware platform, and a code stream is assembled that describes the manipulations required to render these data structures. The compiler

includes a front end configured to read the geometric data and attributes (an art asset) output from a 3D modeling package and shaders in a platform independent form and perform platform-independent optimizations, and a back end configured to perform platform-specific optimizations and generate platform-targeted data structures and code streams.

With respect to amended claim 1, amended claim 1 recites:

> A method comprising:
> identifying a set of commands to be submitted to a processing unit, wherein the set of commands were captured and saved as they were previously submitted to the processing unit;
> selecting a subset of the set of commands;
> submitting the subset of the set of commands to the processing unit for processing; and
> analyzing processing performed by the processing unit in response to the subset of the set of commands.

Applicant respectfully submits that no such method is disclosed by Schenk.

Schenk discusses simulating the contents of the GPU memory over the execution of the rendering of a model, and noting uploads that do not change the memory image (see, ¶ 115). Schenk also discusses that the simulation is only the uploading of the data to the GPU, not GPU execution in detail (see, ¶ 116). Thus, it can be seen that Schenk simulates the contents of the GPU memory based on the uploading of data to the GPU; Schenk does not simulate the detailed execution of the GPU and thus cannot disclose analyzing **processing performed by the processing unit** as recited in claim 1. Schenk is simulating the uploading of data to the CPU and the contents of the GPU memory from those data uploads, not analyzing processing performed by the processing unit in response to the subset of the set of commands as recited in claim 1.

Furthermore, Applicant respectfully submits that there is no discussion or mention in Schenk of a set of commands were captured and saved as they were previously submitted to a processing unit. Schenk discusses simulating the contents of the GPU memory based on the uploading of data to the GPU, not capture and saving of a set of commands as they were previously submitted to the processing unit.

For at least these reasons, Applicant respectfully submits that claim 1 is allowable over Schenk.

With respect to claims 3, 4 and 7, given that claims 3, 4 and 7 depend from claim 1, Applicant respectfully submits that claims 3, 4 and 7 are likewise allowable over Schenk for at least the reasons discussed above with respect to claim 1.

With respect to claim 15, claim 15 recites:

> One or more computer readable media having one or more instructions that, when executed by one or more processors, causes the one or more processors to:
>
> capture a state of a graphics processing unit;
>
> capture a plurality of commands submitted to the graphics processing unit in order to draw a frame of video; and
>
> save both the captured state and the captured plurality of commands.

Applicant respectfully submits that no such capture and save is disclosed by Schenk.

In the July 25, 2005 Office Action at ¶ 10, it was asserted that:

> Schenk describes . . . to capture a state of a graphics processing unit (*model is a collection of state objects (rendering states),* [0054], *render method data elements transferred to the GPU,* [0082]);

It was further asserted at ¶ 10 that:

> Schenk describes that to capture the state of the graphics processing unit (120, Figure 6) is to obtain the state variables (*the front end provides the back end with a list of packets, each of which has an associated render method and set of input data. The input data is converted into the data defined in the variables section of the render method associated with the packet. The result is an instantiated packet. In an instantiated packet, the data for every variable is either known, or an external symbolic reference is known that will resolve to the memory location of that data at run time*, [0101, 0071]).

Applicant respectfully disagrees.

Applicant notes that Schenk discusses a GPU 120 (see, Fig. 6 and ¶ 46). However, Applicant respectfully submits that nowhere does Schenk discuss or mention capturing a state of the GPU 120. Paragraph 54 of Schenk discusses that the runtime environment presents an API to the user for drawing models of primitive objects, and that a model is a collection of geometric primitives and state objects bound to the shaders required to draw them with predetermined effects. However, the mere disclosure of models, geometric primitives, and state objects does not provide any discussion or mention of capturing a state of the GPU 120.

Paragraph 82 of Schenk discusses cutting data into smaller packets based on hardware restrictions that may limit the number data elements that can be transferred to the GPU. However, the mere disclosure of limiting the number of data elements that can be transferred to the GPU 120 does not provide any discussion or mention of capturing a state of the GPU 120. Paragraph 82 discusses data elements that can be transferred to the GPU, not capturing a state of the GPU.

Paragraph 101 of Schenk discusses variable data construction by the back end of the compiler (see also, Fig. 1, and ¶¶ 90-98). However, the mere disclosure of variable data construction as one pass of the back end of the compiler does not provide any discussion or mention of capturing a state of the GPU 120. Paragraph 101 discusses a pass made by the back end of the compiler, not capturing a state of the GPU.

Paragraph 71 of Schenk discusses a name extension mechanism, and that string variables are preferably referred to in the variable name to extend the variable name at compile time. However, the mere disclosure of a name extension mechanism does not provide any discussion or mention of capturing a state of the GPU 120.

Accordingly, Applicant respectfully submits that there is no discussion or mention of capturing a state of the GPU 120 of Schenk, and thus no discussion or mention of to capture a state of a graphics processing unit as recited in claim 15.

For at least these reasons, Applicant respectfully submits that claim 15 is allowable over Schenk.

With respect to claims 17 and 19, given that claims 17 and 19 depend from claim 15, Applicant respectfully submits that claims 17 and 19 are likewise allowable over Schenk for at least the reasons discussed above with respect to claim 15.

Applicant respectfully requests that the §102 rejections be withdrawn.

35 U.S.C. § 103

Claims 2, 9-12, and 18 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Schenk in view of U.S. Patent No. 6,557,167 to Thelen (hereinafter "Thelen"). Applicant respectfully submits that claims 2, 9-12, and 18 are not obvious over Schenk in view of Thelen.

Thelen is directed to an apparatus and method for analyzing performance of a computer program (see, Title). As discussed in the Abstract of Thelen, the computer program is initially executed according to a predefined set of program execution conditions. As the computer program executes, information for each code segment is logged. Using the logged performance data, a graphical representation of the executed computer program is constructed. A user can then formulate ad hoc queries to analyze any desired performance parameters for the computer program by replaying how the computer program ran using the graphical representation of the executed computer program. The present invention thus allows a user to detect via queries complex performance bottlenecks that are caused by interactions between multiple code segments.

With respect to claim 2, claim 2 depends from amended claim 1 and Applicant respectfully submits that claim 2 is allowable over Schenk for at least the reasons discussed above with respect to amended claim 1. Thelen is not cited as curing, and does not cure, all of the deficiencies of Schenk discussed above with respect to amended claim 1. For at least these reasons, Applicant respectfully submits that claim 2 is allowable over Schenk in view of Thelen.

With respect to amended claim 9, amended claim 9 recites:

> One or more computer readable media having one or more instructions that, when executed by one or more processors, causes the one or more processors to:
>
> modify a stream of commands that were captured and saved as they were previously submitted to a processing unit;
>
> submit the modified stream of commands to the processing unit; and
>
> determine a difference between a first amount of time required by the processing unit to process the stream of commands and a second amount of time required by the processing unit to process the modified stream of commands.

Applicant respectfully submits that no such modification, submission, and determination is disclosed or suggested in Schenk in view of Thelen.

Applicant respectfully submits that there is no discussion or mention in Schenk or Thelen of a stream of commands that were captured and saved as they were previously submitted to a processing unit, much less of modification of such a stream of commands. Schenk discusses simulating the contents of a GPU memory based on the uploading of data to the GPU, not a stream of commands that were captured and saved as they were previously submitted to a processing unit. Thelen discusses replaying a computer program by stepping through the graphical representation of the program (see, col. 3, line 67 – col. 4, line 5), not a stream of commands that were captured and saved as they were previously submitted to a processing unit.

For at least these reasons, Applicant respectfully submits that amended claim 9 is allowable over Schenk in view of Thelen.

With respect to claims 10 and 11, given that claims 10 and 11 depend from amended claim 9, Applicant respectfully submits that claims 10 and 11 are likewise allowable over Schenk in view of Thelen for at least the reasons discussed above with respect to amended claim 9.

With respect to claim 12, claim 12 depends from amended 9 and Applicant respectfully submits that claim 12 is allowable over Schenk in view of Thelen for at least the reasons discussed above with respect to amended claim 9. Furthermore, claim 12 recites:

> One or more computer readable media as recited in claim 9, wherein to modify the stream of commands is to change one or more instructions of an internal program of the processor to reveal a value of an internal variable of the internal program.

Applicant respectfully submits that no such change of one or more instructions of an internal program of the processor to reveal a value of an internal variable of the internal program is recited in Schenk in view of Thelen.

In the July 25, 2005 Office Action at ¶ 21, Schenk at paragraph 67 was cited as disclosing this change of claim 12. Paragraph 67 of Schenk reads as follows:

> **[0067]** Not all variables used by a shader can be made available through a compiler preprocess, however. For example, data such as transformation matrices are not available at compile time. They could be communicated as implicit variables. This would restrict the user from extending the set of such variables. According to one aspect, the render method specification is extended through external linkage. By adding the extern keyword to a variable declaration along with an assignment of the form=variable_name, a reference is made to an external variable named variable_name. In one example in FIG. 2, the runtime is responsible for replacing unresolved references to the variable named Viewport::XFormProject with a pointer to the actual runtime address of this variable. This external reference is resolved when the asset is loaded. Assets are preferably stored in ELF format (e.g., Tool Interface Standards, 1998, ELF: Executable and linkable format. ftp://ftp.intel.com/pub/tis), and provide the external linkage through an ELF dynamic loader. The library registers a number of variables with the dynamic loader, making transformation matrices, for example, available to render methods. The user may also, at runtime, register his own variables.

This cited portion of Schenk discusses that by adding the extern keyword to a variable declaration along with an assignment of the form=variable_name, a reference is made to an external variable named variable_name, and that the external reference is resolved when the asset is loaded. Nowhere in this cited portion of Schenk is there any discussion or mention of a reference being made to an external variable so that the internal variable is revealed as asserted in the July 25, 2005 Office Action. This cited portion of Schenk discusses resolving an external reference when the asset is loaded, not revealing the value of an internal variable of an internal program of a processor. Accordingly, Applicant respectfully submits that Schenk does not disclose or suggest to modify the stream of commands is to change one or more instructions of an internal program of the processor to reveal a value of an internal variable of the internal program as recited in claim 12. With respect to Thelen, Thelen is not cited as curing, and does not cure, these deficiencies of Schenk.

For at least these reasons, Applicant respectfully submits that claim 12 is allowable over Schenk in view of Thelen.

With respect to claim 18, claim 18 depends from claim 15 and Applicant respectfully submits that claim 18 is allowable over Schenk for at least the reasons discussed above with respect to claim 15. Thelen is not cited as curing, and does not cure, all of the deficiencies of Schenk discussed above with respect to claim 15. For at least these reasons, Applicant respectfully submits that claim 18 is allowable over Schenk in view of Thelen.

Claims 6 and 13 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Schenk in view of Thelen and further in view of U.S. Patent No.

5,970,439 to Levine et al. (hereinafter "Levine"). Applicant respectfully submits that claims 6 and 13 are not obvious over Schenk in view of Thelen and further in view of Levine.

Levine is directed to performance monitoring in a data processing system (see, Title). As discussed in the Abstract of Levine, performance monitoring capabilities are expanded to an entire data processing system so that performance analyses can be made for operations occurring within the entire data processing system and not merely within the processor or any other device containing the performance monitor. Therefore, there is a provision for communicating performance monitor-related signals between the various performance monitors within the various devices and processor within a data processing system

With respect to claim 6, claim 6 recites:

> A method as recited in claim 1, further comprising:
> setting the processing unit, prior to submitting the subset of the set of commands to the processing unit, to a particular state, wherein the particular state is a same state as the processing unit was in at the time capture of the set of commands began.

Applicant respectfully submits that no such setting is disclosed or suggested by Schenk in view of Thelen and further in view of Levine.

In the July 25, 2005 Office Action at ¶ 24, Levine was cited as disclosing this setting of claim 6. Applicant respectfully disagrees. Levine discusses that performance monitoring data is collected (see, col. 10, lines 25-27). Levine also discusses that the collected data is analyzed to identify potential areas of system enhancements (see, col. 10, lines 54-56). However, Applicant respectfully submits that nowhere does Levine disclose or suggest setting the processing unit to a particular state that is a same state as the processing unit was in at the time

capture of the set of commands began. Levine discusses analyzing collected data, not setting the processing unit back to a state identified by that collected data.

Applicant notes that Levine states that "From sequencer unit 18, branch unit 20 inputs instructions and signals indicating a present state of processor 10." (see, col. 5, lines 49-51). However, nowhere in Levine is there any discussion or mention that this present state of processor 10 is a state that the processor had when capture of commands began. Furthermore, it is not clear from this statement in Levine what the instructions and signals indicating the present state of processor 10 are used for – there is no discussion or mention in Levine that these instructions and signals are used to set the processor, much less that they are used to set the processor to a particular state that is a same state as the processor was in at the time capture of the set of commands began.

With respect to Schenk and Thelen, Schenk and Thelen are not cited as curing, and do not cure, these deficiencies of Levine.

For at least these reasons, Applicant respectfully submits that claim 6 is allowable over Schenk in view of Thelen and further in view of Levine.

With respect to claim 13, Applicant respectfully submits that, similar to the discussion above regarding claim 6, Schenk in view of Thelen and further in view of Levine does not disclose or suggest to set the processing unit, prior to submission of the modified stream of commands to the processing unit, to a particular state, wherein the particular state is a same state as the processing unit was in at the time capture of the stream of commands began as recited in claim 13. For at least these reasons, Applicant respectfully submits that claim 13 is allowable over Schenk in view of Thelen and further in view of Levine.

Claims 8 and 14 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Schenk in view of Thelen and U.S. Patent No. 6,446,029 to Davidson et al. (hereinafter "Davidson"). Applicant respectfully submits that claims 8 and 14 are not obvious over Schenk in view of Thelen and Davidson.

Davidson is directed to providing temporal threshold support during performance monitoring of a pipelined processor (see, Title). As discussed in the Abstract of Davidson, the processor may contain a performance monitor for monitoring for the occurrence of an event within a data processing system. An event to be monitored may be specified through software control, and the occurrence of the specified event is monitored during the execution of an instruction in the execution pipeline of the processor. A particular instruction may be specified to execute within a threshold time for each stage of the instruction pipeline. The specified event may be the completion of a single tagged instruction beyond the specified threshold interval for a stage of the instruction pipeline. The performance monitor may contain a number of counters for counting multiple occurrences of specified events during the execution of multiple instructions, in which case the specified events may be the completion of tagged instructions beyond a threshold interval for any stage of the multiple stages of the execution pipeline. As the instruction moves through the processor, the performance monitor collects the events and provides the events for optimization analysis.

With respect to claim 8, claim 8 depends from amended claim 1 and Applicant respectfully submits that claim 8 is allowable over Schenk for at least the reasons discussed above with respect to amended claim 1. Thelen and Davidson are not cited as curing, and do not cure, all of the deficiencies of Schenk

discussed above with respect to amended claim 1. For at least these reasons, Applicant respectfully submits that claim 8 is allowable over Schenk in view of Thelen and Davidson.

With respect to claim 14, claim 14 depends from amended claim 9 and Applicant respectfully submits that claim 14 is allowable over Schenk in view of Thelen for at least the reasons discussed above with respect to amended claim 9. Davidson is not cited as curing, and does not cure, all of the deficiencies of Schenk and Thelen discussed above with respect to amended claim 9. For at least these reasons, Applicant respectfully submits that claim 14 is allowable over Schenk in view of Thelen and Davidson.

Claim 16 stands rejected under 35 U.S.C. §103(a) as being unpatentable over Schenk in view of U.S. Patent Application Publication No. 20030232648A1 to Prindle (hereinafter "Prindle"). Applicant respectfully submits that claim 16 is not obvious over Schenk in view of Prindle.

Prindle is directed to videophone and videoconferencing apparatus and method for a video game console (see, Title). As discussed in the Abstract of Prindle, Prindle discusses a videophone and videoconferencing system for multiple video game consoles residing on a network. Each client video game console has a camera and microphone attached, client software executing on the game console, that can be controlled by the video game controller through a Graphical User Interface (GUI), which is displayed from the client video game console on a standard television set. Each console comprises a networked interface card (NIC) or modem, a network connection, and software executing on the client video game console that can establish peer-to-peer or client-to-server

network connections. Each console supports the H.323 standard for conferencing, and can send and receive, and encode and decode video and audio signals, and display that and other data on screen. This system and apparatus allows a game console to become a simple and cost-effective method for videophone and videoconferencing between client video game consoles and other client computers, and in one implementation, can be used as a peer-to-peer videophone for a networked game console.

As discussed at MPEP §§ 2142 and 2143, to establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Applicant respectfully submits that there is no suggestion or motivation to combine Schenk and Prindle, and thus that no *prima facie* case of obviousness of claim 16 has been established. Schenk, as discussed above, is directed to implementing shader-driven compilation of rendering assets. Prindle, as discussed above, is directed to a videophone and videoconferencing system for a video game console. There is no suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to

combine the shader-driven compilation of rendering assets teachings of Schenk with the video game console videoconferencing of Prindle.

In the July 25, 2005 Office Action at ¶ 29, it was asserted that:

> It would have been obvious to one of ordinary skill in this art at the time of invention by application to modify the device of Schenk so that the request is received from a remote computing device as suggested by Prindle because Prindle suggests that this is needed in order to perform real-time videoconferencing.

Applicant respectfully disagrees. The need for a remote computing device in Prindle in order to perform the videoconferencing of Prindle does not provide any motivation to combine that videoconferencing with the shader-driven compilation of rendering assets of Schenk.

As discussed in MPEP §2143.01, the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. Accordingly, Applicant respectfully submits that the mere fact that Schenk and Prindle can be combined does not render the resultant combination obvious because the prior art does not suggest the desirability of the combination.

Furthermore, assuming for the sake of argument that Schenk and Prindle were combined, the combination still would not disclose or suggest to perform the captures and save in response to a request to capture the frame, wherein the request is received from a remote computing device as recited in claim 16. Claim 16 depends from claim 15 and Applicant respectfully submits that claim 16 is allowable over Schenk for at least the reasons discussed above with respect to claim 15. Prindle is not cited as curing, and does not cure, all of the deficiencies of Schenk discussed above with respect to claim 15. For at least these reasons,

Applicant respectfully submits that claim 16 is allowable over Schenk in view of Prindle.

Claim 20 stands rejected under 35 U.S.C. §103(a) as being unpatentable over Schenk in view of U.S. Patent No. 6,173,368 to Krueger et al. (hereinafter "Krueger"). Applicant respectfully submits that claim 20 is not obvious over Schenk in view of Krueger.

Krueger is directed to a class categorized storage circuit for storing non-cacheable data until receipt of a corresponding terminate signal (see, Title). As discussed in the Abstract of Krueger, Krueger discusses a microprocessor (62) for coupling to an external read/write memory (70) having an addressable storage space for storing data. The microprocessor includes a data storage circuit (76) for storing a portion of the data, where that portion of data comprises non-cacheable data. The microprocessor further includes a class storage circuit (80) for storing a class identifier corresponding to the portion of the non-cacheable data, as well as an input (TERMINATE) for receiving a terminate signal and an input (CLASS) for receiving a class signal. Lastly, the microprocessor includes an indicator (82) for indicating that the portion of the non-cacheable data in the data storage circuit is expired in response to assertions of the terminate signal and the class signal matching the class identifier.

Applicant respectfully submits that there is no suggestion or motivation to combine Schenk and Krueger, and thus that no *prima facie* case of obviousness of claim 20 has been established. Schenk, as discussed above, is directed to implementing shader-driven compilation of rendering assets. Krueger, as discussed above, is directed to a circuit for storing non-cacheable data. There is

no suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to combine the shader-driven compilation of rendering assets teachings of Schenk with the storage circuit of Krueger.

As discussed in MPEP §2143.01, the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. Accordingly, Applicant respectfully submits that the mere fact that Schenk and Krueger can be combined does not render the resultant combination obvious because the prior art does not suggest the desirability of the combination.

Furthermore, assuming for the sake of argument that Schenk and Krueger were combined, the combination still would not disclose or suggest the elements of claim 20. Claim 20 depends from claim 15 and Applicant respectfully submits that claim 20 is allowable over Schenk for at least the reasons discussed above with respect to claim 15. Krueger is not cited as curing, and does not cure, all of the deficiencies of Schenk discussed above with respect to claim 15. For at least these reasons, Applicant respectfully submits that claim 20 is allowable over Schenk in view of Krueger.

Applicant respectfully requests that the §103 rejections be withdrawn.

Conclusion

Claims 1-4 and 6-20 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter in this case remain unresolved, the undersigned attorney respectfully

requests a telephone conference with the Examiner to resolve any such outstanding matter.

Respectfully Submitted,

Date: 10/25/05

By: _____
Allan T. Sponseller
Reg. No. 38,318
(509) 324-9256